

Soru	1	2 / 2	3	4	5	6	7	8	Toplam
Değer	5	10 / 10	25	10	20	10	10	10	110
Puan									

1. (5 puan) Aşağıdaki kısaltmaları açık olarak yazın:

- RDF:
- RDFa:
- W3C:
- SPARQL:
- AAA:

2. (RDFa, 10 puan) Aşağıda bir HTML kod parçası verilmiştir. Bu dökümanda saklı semantik veriyi üçlüler (triple) olarak alt alta yazınız.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
  "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  version="XHTML+RDFa 1.0" xml:lang="en">
  <head>
    <title>John's Home Page</title>
    <base href="http://example.org/john-d/" />
    <meta property="dc:creator" content="Jonathan Doe" />
    <link rel="foaf:primaryTopic" href="http://example.org/john-d/#me" />
  </head>
  <body about="http://example.org/john-d/#me">
    <h1>John's Home Page</h1>
    <p>My name is <span property="foaf:nick">John D</span> and I like
      <a href="http://www.neubauten.org/" rel="foaf:interest"
        xml:lang="de">Einstürzende Neubauten</a>.
    </p>
    <p>
      My <span rel="foaf:interest" resource="urn:ISBN:0752820907">favorite
        book is the inspiring <span about="urn:ISBN:0752820907"><cite
          property="dc:title">Weaving the Web</cite> by
          <span property="dc:creator">Tim Berners-Lee</span></span>
        </span>
      </p>
  </body>
</html>
```

Cevap:

N3 formatında:

```

http://example.org/john-d/      dc:creator      "Jonathan Doe";
                                primaryTopic    http://example.org/john-d/#me.
http://example.org/john-d/#me  foaf:nick      "John D";
                                foaf:interest    http://www.neubauten.org;
                                foaf:interest    urn:ISBN:0752820907.
urn:ISBN:0752820907             dc:creator      "Tim Berners-Lee";
                                dc:title         "Weaving the Web".

```

RDF/XML formatında:

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description rdf:about="http://example.org/john-d/">
    <dc:creator xml:lang="en">Jonathan Doe</dc:creator>
    <foaf:primaryTopic>
      <rdf:Description rdf:about="http://example.org/john-d/#me">
        <foaf:nick xml:lang="en">John D</foaf:nick>
        <foaf:interest rdf:resource="http://www.neubauten.org/" />
        <foaf:interest>
          <rdf:Description rdf:about="urn:ISBN:0752820907">
            <dc:creator xml:lang="en">Tim Berners-Lee</dc:creator>
            <dc:title xml:lang="en">Weaving the Web</dc:title>
          </rdf:Description>
        </foaf:interest>
      </rdf:Description>
    </foaf:primaryTopic>
  </rdf:Description>
</rdf:RDF>

```

2. (RDF, 10) Aşağıda verilen ilişkisel modele uygun verileri (tablolar), RDF veri modeline dönüştürünüz. Sonucu üçlüler olarak ifade ediniz (*herhangi bir formatta olabilir - turtle, n3, rdf/xml gibi*)

Ders

kod	dersAdı	hoca
BİL546	Semantik Web	3
BİL553	Algoritma	2

Hoca

no	ad
1	Murat
2	Fatih
3	Erdoğan

Cevap:

```

:BİL546      rdf:type      :Ders.
:BİL553      rdf:type      :Ders.
:BİL546      :kod          "BİL546" ;
              :dersAdı     "Semantik Web" ;
              :hoca        :hoca3.
:BİL553      :kod          "BİL553" ;
              :dersAdı     "Algoritma" ;
              :hoca        :hoca2.
:hoca1       rdf:type      :Hoca ;
              :no          "1" ;
              :ad          "Murat".
:hoca2       rdf:type      :Hoca ;
              :no          "2" ;
              :ad          :Murat.
:hoca3       rdf:type      :Hoca ;
              :no          "3" ;
              :ad          "Erdoğan" .

```

3. SPARQL, 25) Aşağıdaki RDF veri modeli veriliyor. İlişkiler propertyName(Class) şeklinde verilmiştir. Örneğin açılanDers(Ders) açılan bir dersin hangi ders olduğunu (Ders) gösterir.

Hoca(no, ad)

Ders(kod, dersAdı, kredi)

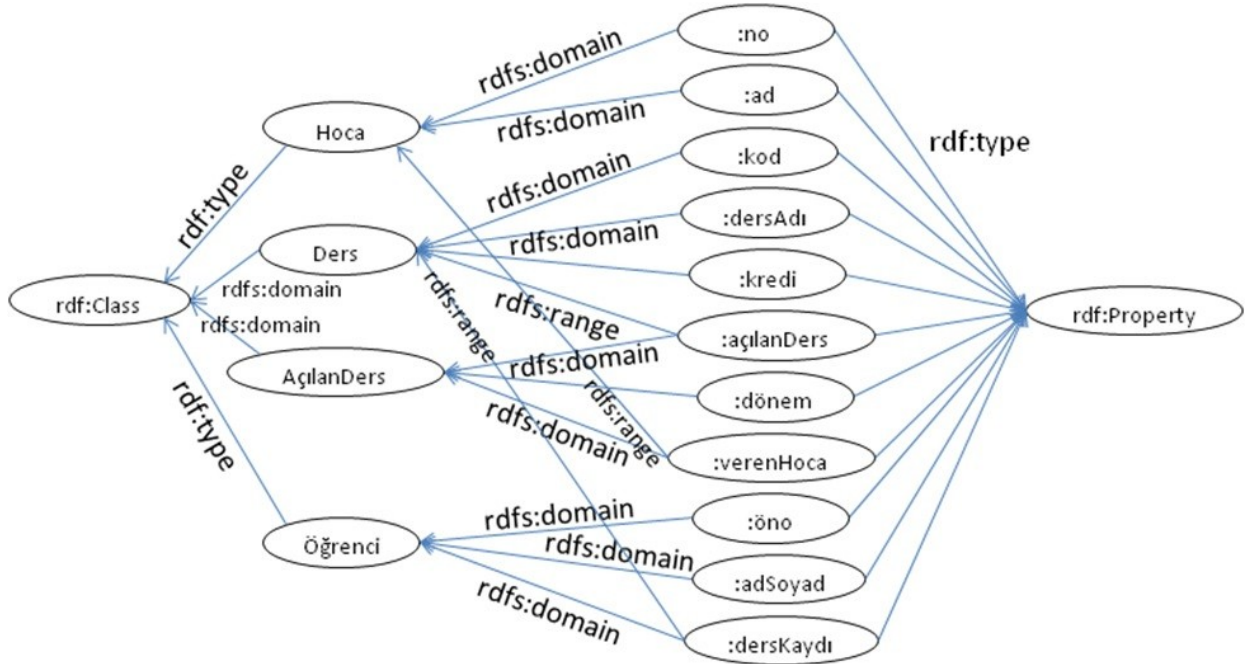
AçılanDers(açılanDers(Ders), dönem, verenHoca(Hoca))

Öğrenci(öno, adSoyad, dersKaydı(Ders)*) * : birden fazla ders kaydını gösterir

Buna göre

- a. Yukarıdaki veri modelini çizge (graph) olarak gösteriniz.

Cevap: (not: dersKaydı(Ders)* yanlış, dersKaydı(AçılanDers)* olmalı, fakat değiştirilmedi)



Aşağıdaki sorguları SPARQL ile yazınız:

- b. “2011g” döneminde açılan derslerin kodu, adı ve dersi veren hocanın ismi.

Cevap:

```
SELECT { ?kod ?dersAdı ?ad }
WHERE {
  ?d a :Ders ;
  ?a a :AçılanDers ;
  :açılanDers ?d ;
  :dönem "2011g" ;
  :verenHoca ?h .
  ?h :ad ?ad.
  ?d :kod ?kod ;
  :dersAdı ?dersAdı.
}
```

- c. “Algoritma” dersini daha önce veren hocalarımızın isimleri (bir kere).

Cevap:

```
SELECT DISTINCT ?ad
WHERE {
  ?d a :Ders ;
  ?a a :AçılanDers ;
```

```

        :açılanDers ?d ;
        :verenHoca ?h .
    ?h :ad ?ad .
    ?d :kod ?kod ;
        :dersAdı "Algoritma".
}

```

- d. Hocalarımızın her dönem verdikleri ders sayısı (*adSoyad, dönem, ders sayısı*) (adSoyad ve dönem sırasında).

Cevap:

```

SELECT ?ad ?dönem COUNT(?a)
WHERE {
    ?d a :Ders ;
    ?a a :AçılanDers ;
        :dönem ?dönem ;
        :açılanDers ?d ;
        :verenHoca ?h .
    ?h :ad ?ad .
    ?d :kod ?kod ;
        :dersAdı "Algoritma".
}
GROUP BY {?adSoyad ?dönem}
ORDER BY {?adSoyad ?dönem}

```

- e. "2011g" döneminde ders vermeyen hocalarımızın isimleri.

Cevap:

```

SELECT ?ad
WHERE {
    ?a a :AçılanDers ;
        :dönem ?dönem ;
        :açılanDers ?d ;
        :verenHoca ?h .
    UNSAID { ?a :dönem "2011g" }
    ?h :ad ?ad .
}

```

- f. "2011g" döneminde BİL546 ve BİL553 derslerinden herhangi birini veya ikisini alan öğrencilerin adları (*bir kere*).

Cevap:

```

SELECT ?adSoyad
WHERE {
    ?a a :AçılanDers ;
        :dönem "2011g" ;
        :açılanDers ?d ;
        :verenHoca ?h .
    ?d :kod ?kod.
    ?o :dersKaydı ?d ;
        :adSoyad ?adSoyad.
    FILTER (?kod="BİL553" || ?kod="BİL546")
}

```

4. (Inferencing/SPARQL, 10 puan) Gerçek hayatta kullanımlarına bağlı kalınarak aşağıdaki sınıf ve ilişkiler tanımlı olsun.

Sınıflar: **Kişi, Erkek, Kadın**

İlişkiler: **kardeşi, babası, annesi**

Yalnızca yukarıdaki tanımları kullanarak “**dayısı**” ilişkisini (“Elif’in dayısı Mehmet’tir” gibi) SPARQL CONSTRUCT kullanarak tanımlayınız.

Cevap:

```
CONSTRUCT { :a :dayısı :c }
WHERE {
    :a :annesi :b
    :b :kardeşi :c
    :c rdf:type :Erkek
}
```

5. (Inferencing, 20 puan) Aşağıdaki bilgiler veriliyor (assertions)

:p1 rdfs:subPropertyOf :p2 (1)

:p2 rdfs:domain :C1 (2)

:p2 rdfs:range :C2 (3)

:C1 rdfs:subClassOf :C3 (4)

:p1 rdf:type owl:FunctionalProperty (5)

:a :p1 :b (6)

:a :p1 :c (7)

Bu bilgilerden çıkarabileceğiniz (inferencing) tüm bilgileri çıkarın ve üçlüler olarak aşağıda listeleyin.

Cevap:

:a :p2 :b (1)

:a :p2 :c (1)

:a rdf:type :C1 (2)

:b rdf:type :C2 (3)

:c rdf:type :C2 (3)

:a rdf:type :C3 (4)

:b owl:sameAs :c (5)

6. (OWL, 10 puan) **owl:ObjectProperty** ve **owl:DataProperty** arasında ne fark vardır? Birer kullanım örneği veriniz.

Cevap:

owl:ObjectProperty'nin object'i veya range'i her zaman bir resource veya kaynaktır. Dolayısıyla owl:ObjectProperty iki kaynağı (resource) ilişkilendirir.

```
owl:Class ← rdfs:domain- owl:ObjectProperty -rdfs:range→ owl:Class
```

Örnek: :BİL546 :verenHoca :hoca1

owl:DataProperty ise bir resource'u (subject) bir literal (XML türünde) değerle ilişkilendirir.

```
owl:Class ← rdfs:domain- owl:DataProperty -rdfs:range→ XML data
```

Örnek: :BİL546 :kod "BİL546"

7. (OWL, 10 puan) Aşağıdakileri yazın:

- a. Bir property **:p**'nin owl:TransitiveProperty türünde olmasını SPARQL CONSTRUCT sorgusu ile tanımlayınız (inference kuralı SPARQL ile yazılacak).

Cevap:

```
CONSTRUCT {?x ?p ?z .}
WHERE {?x ?p ?y .
       ?y ?p ?x .
       ?p a owl:TransitiveProperty . }
```

- b. Property **p**'nin owl:SymmetricProperty olmasını SPARQL CONSTRUCT sorgusu ile tanımlayınız (inference kuralı SPARQL ile yazılacak).

Cevap:

```
CONSTRUCT {?y ?p ?x .}
WHERE {?x ?p ?y .
       ?p a owl:SymmetricProperty . }
```

8. (OWL, 10 puan) Aşağıdaki soruları cevaplayınız:

- a. İki sınıfın (class) aynı olduklarını hangi OWL property'si ile ifade edersiniz? _____
- b. İki ilişkinin (property) aynı olduklarını hangi OWL property'si ile ifade edersiniz? _____
- c. İki nesnenin (instance) aynı olduklarını hangi OWL property'si ile ifade edersiniz? _____

Cevap:

- a. owl:equivalentClass
- b. owl:equivalentProperty
- c. owl:sameAs